

-----  
**First name and last name**

-----  
**Student number**

Fill out with capital letters

Writing time: 75 minutes, date: November 19, 2019

Comments: in case of all programs assume that libraries iostream and stdlib are attached and the namespace std is available. Only places marked for the answers are graded. In case you find a mistake or ambiguity in a question, please write an appropriate comment which explains that. The number of marks for the test is 0-100 points (passing threshold = 50%).

**Question 1. (20 pts. = 2\*10 pts.)**

What numbers will be returned by the following calls to the function *f*?

char x1[] = { "bit" };  
char x2[] = { "program" };

Answer:

f( x1 ) = \_\_\_\_\_

f( x2 ) = \_\_\_\_\_

```
int f( char *s ) {  
    int z = 0;  
    while ( *(s+z) != '\0' )  
        z++;  
    *(s+z/2) = '\0';  
    if ( z <= 1 )  
        return z+3;  
    else  
        return f( s ) + f( s + z/2+1 );  
}
```

**Question 2. (21 pts. = 3\*7 pts.)**

Write the content of the structure *s* right before the end of the execution of the function *main*.

Answer:

s.x = \_\_\_\_\_

s.y = \_\_\_\_\_

s.z = \_\_\_\_\_

```
typedef struct {  
    int x, y, z;  
} type_t;  
void f( type_t x ) { x.x = 1; }  
void g( type_t *x ) { x->y = 2; f( *x ); }  
void h( type_t **x ) { (*x)->z = 3; }  
int main() {  
    type_t s = { 0, 0, 0 }, *ps = &s;  
    f(s);  
    g( &s );  
    h( &ps );  
    return 0;  
}
```

**Question 3. (19 pts.)**

What the following program will write to the screen?

Answer: \_\_\_\_\_

```
#define SIZE 12  
int main() {  
    char s[SIZE], str[] = { "programming" };  
    int i=0, j=5;  
  
    while ( str[i] != '\0' ) {  
        s[j--] = str[i++];  
        if ( j < 0 )  
            j = SIZE - 1;  
    }  
    s[j] = str[i];  
    cout << s;  
    return 0;  
}
```

**Question 4. (20 pts. = 5\*4 pts.)**

Give the text that is written to the screen as a result of executing the subsequent instructions *cout* (in place for an answer marked with label "Instruction x:" write the text that is printed to the screen by instruction *cout* marked with comment */\* I-x \*/*. Write *ERR* if the answer cannot be uniquely determined. Binary encoding of number is assumed, as presented during the lectures, i.e., U2. If some instruction results in an execution error, then also write *ERR* as an answer and continue your analysis by skipping this instruction.

Answers:

Instruction 1: \_\_\_\_\_

Instruction 2: \_\_\_\_\_

Instruction 3: \_\_\_\_\_

Instruction 4: \_\_\_\_\_

Instruction 5: \_\_\_\_\_

```
struct a_t { char c; int i; } s;
int tab[] = { 3, 4, 5, 6 };
int main() {
    int *p = &(tab[1]), *p1 = tab + 4;

    cout << (13 ^ 7); /* I-1 */

    cout << (sizeof(s.c)+sizeof(s.i)) /
            (sizeof(s)); /* I-2 */

    cout << p1-p; /* I-3 */

    cout << 15/2/((float)2); /* I-4 */

    cout << (sizeof tab)/(sizeof p); /* I-5 */
    return 0;
}
```

**Question 5. (20 pts. = 5\*4 pts.)**

Fill out the field so that the program compiles successfully, executes without errors and prints to the screen:

**15**

Memory allocations should be done in such a way that each call to *malloc* returns the address to the smallest array possible ensuring correct program execution. We assume that during the execution of the program each call to *malloc* returns a pointer different than *NULL*.

```
#define SIZE 6

_____ f( int tab[], int z ) {

    int *x = (int *) malloc( _____ );
    int i, j=0;
    for ( i=0; i < z; i++ )

        if ( _____ )
            x[j++] = tab[i];

    while ( j < z )

        x[_____] = 0;

    return _____;
}

int main() {
    int z[] = { -2, 3, 5, -1, 4, -3, 2, -4 }, *w[2], d, i;

    w[0] = f( z, 2 );
    d = w[0][0]+w[0][1];
    w[1] = f( z, SIZE );
    for ( i=0; i < SIZE; i++ )
        d += w[1][i];
    cout << d;
    free( w[0] ); free( w[1] );
    return 0;
}
```